



Complétude des extensions en arbres de théories

Khalil Djelloul, Thi-Bich-Hanh Dao

► To cite this version:

Khalil Djelloul, Thi-Bich-Hanh Dao. Complétude des extensions en arbres de théories. Journées Francophones de Programmation par Contraintes, 2006, Nîmes - Ecole des Mines d'Alès. inria-00085774

HAL Id: inria-00085774

<https://inria.hal.science/inria-00085774>

Submitted on 14 Jul 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Complétude des extensions en arbres de théories

Khalil Djelloul ¹

Thi-Bich-Hanh Dao ²

¹ Laboratoire d'Informatique Fondamentale de Marseille, France.

² Laboratoire d'Informatique Fondamentale d'Orléans, France.

khalil.djelloul@lif.univ-mrs.fr Dao@univ-orleans.fr

Résumé

Nous présentons dans ce papier une méthode pour combiner une théorie T quelconque du premier ordre avec la théorie des arbres éventuellement infinis. Sémantiquement cette nouvelle théorie hybride n'est rien d'autre qu'une axiomatisation de l'extension en arbres des éléments des modèles de la théorie T . Tout d'abord, ayant une axiomatisation d'une théorie T , nous donnons l'axiomatisation de la théorie T^* de l'extension en arbre de T et présentons son modèle standard M^* . Nous introduisons ensuite une nouvelle classe de théories dite *flexibles* et montrons que si T est flexible alors T^* est complète. Les théories flexibles sont des théories ayant des propriétés élégantes qui nous permettent de manipuler aisément les formules. Enfin nous présentons un algorithme de décision de propositions dans T^* pour toute théorie T flexible. L'algorithme est donné sous forme d'un ensemble de six règles de réécriture qui pour toute proposition φ donnent soit *vrai* soit *faux*.

Abstract

We present in this paper a general way to combine a first order theory T with the theory of finite or infinite trees. Semantically this new hybrid theory is nothing other than an axiomatization of the extension into trees of the elements of the models of the theory T . First of all, having an axiomatization of a theory T we give the axiomatization of the theory T^* of the extension into trees of T and present its standard model M^* . Then, we introduce a new class of theories that we call *flexible* and show that if T is flexible then T^* is complete. The flexible theories are theories that have elegant properties which enable us to handle easily formulas. Finally we give an algorithm of decision of propositions in T^* for every flexible theory T . This algorithm is given in the form of six rewriting rules which for every propositions φ give either true or false.

1 Introduction

Les arbres éventuellement infinis jouent un rôle fondamental en programmation. Rappelons qu'Alain Colmerauer a modélisé le fonctionnement de Prolog II, III et IV par la résolution d'équations et d'inégalités dans les arbres infinis [1, 2]. Il a tout d'abord intégré l'unification des termes infinis dans Prolog II, ainsi qu'un prédicat de non-égalité. Il a ensuite ajouté dans Prolog III le domaine des rationnels munis de l'addition, de la soustraction et d'une relation d'ordre dense sans extrême et a donné une technique pour tester la satisfaisabilité d'un système d'équations, d'inéquations et d'inégalités sur un mélange harmonieux d'arbre infinis et de rationnels. Quant à Prolog IV, des notions de listes, d'intervalles et de booléens ont été ajoutées. Récemment, Alain Colmerauer et Thi-Bich-Hanh Dao ont également montré qu'une extension de Prolog IV est possible en permettant à l'utilisateur de quantifier existentiellement ou universellement les clauses de Prolog II. Ils ont donc donné un algorithme de résolution de contraintes générales du premier ordre dans la théorie des arbres finis et éventuellement infinis [4, 3], en utilisant notamment un des résultats de Michael Maher qui a donné et justifié des axiomatisations complètes de différents ensemble d'arbres munis d'un ensemble infini de symboles de fonction [7].

Quant à nous, nous présentons dans ce papier une continuité d'idée d'extension du modèle de Prolog en donnant d'abord une manière automatique pour combiner une théorie T du premier ordre avec la théorie des arbres finis ou infinis, puis un algorithme de décision de propositions quelconques dans cette théorie hybride. L'une des difficultés majeures de cette combinaison réside dans le fait que la théorie T et la théorie des arbres éventuellement infini peuvent avoir des signa-

tures non disjointes, c'est-à-dire l'existence d'au moins un symbole de fonction ayant deux comportements différents selon que l'on soit dans la théorie des arbres ou dans la théorie T . Il faut donc d'abord trouver un sens sémantique à cette combinaison puis donner une axiomatisation harmonieuse de ce mélange. Pour cela, nous définissons sémantiquement cette combinaison comme étant une extension en arbre des éléments des modèles de la théorie T . Ainsi, l'axiomatisation de l'extension en arbres de T , notée T^* , se fait essentiellement à partir des trois axiomes modifiés de Michael Maher sur la théorie des arbres éventuellement infinis [7] et de l'axiomatisation de la théorie T en introduisant des contraintes de typage. Nous donnons également une définition formelle du modèle standard M^* de la théorie T^* . Pour montrer la complétude de la théorie T^* , nous introduisons une nouvelle classe de théories que nous appelons *flexible* et montrons que si T est une théorie flexible alors T^* est une théorie complète. Les théories flexibles sont des théories ayant des propriétés agréables qui nous permettent de manipuler facilement les formules du premier ordre. Nous donnons également dans ce papier un algorithme de décision de propositions (formules sans variables libres) dans T^* pour toute théorie T flexible. Cet algorithme est donné sous forme de six règles de réécriture qui pour toute proposition φ donnent soit *vrai* soit *faux*.

Il existe plusieurs applications pratiques des théories étendues en arbres. Notons que ces théories sont dotées d'un très grand pouvoir d'expression et permettent de modéliser sous forme de contraintes de premier ordre des problèmes qui utilisent des constructions d'arbres et dont les feuilles s'évaluent par d'autres théories. On citera par exemple les contraintes modélisant les positions k -gagnantes des jeux à deux partenaires utilisées par A. Colmerauer et B. Dao [4, 3] qui se modélisent plus facilement en utilisant une extension en arbres des entiers additifs ordonnés. L'idée d'extension du modèle de Prolog par d'autres théories utilise également cette notion d'extension en arbres de théories.

Ce papier est organisé en cinq sections suivies d'une conclusion. Cette introduction est la première section. Dans la section 2, nous rappelons les définitions de base de signature, modèle, théorie et quantificateur vectoriel. Nous présentons également un quantificateur particulier appelé *zéro-infin* ainsi qu'une condition suffisante de complétude de théories du premier ordre. Ces deux notions sont définies en détails dans [5]. Dans la section 3, nous présentons l'axiomatisation T^* de l'extension en arbres de toute théorie du premier ordre T . Nous introduisons ensuite une nouvelle classe de théories dite *flexible* et montrons que si T est flexible alors l'extension en arbres de T , c'est-à-dire T^* , est complète. Dans la section 4, nous mon-

trons que toute théorie qui satisfait notre condition de complétude, admet un algorithme de décision de propositions. Ainsi, pour toute théorie flexible T , il existe un algorithme de décision de propositions dans T^* qui pour toute proposition φ donne soit *vrai* soit *faux*. Enfin, dans la section 5, nous présentons la théorie T_{ad}^* de l'extension en arbre des rationnel additifs ordonnés T_{ad} . Nous montrons alors que T_{ad} est flexible et par conséquent que T_{ad}^* est complète. L'axiomatisation de T^* , les théories flexibles, la preuve de complétude de T^* pour toute théorie flexible T ainsi que l'algorithme de décision sont nos contributions essentielles dans ce papier.

2 Préliminaires

Soient V un ensemble infini dénombrable de *variables* et L l'ensemble des symboles *logiques*

$$=, \text{vrai}, \text{faux}, \neg, \wedge, \vee, \rightarrow, \leftrightarrow, \forall, \exists, (,).$$

Soit maintenant un ensemble supplémentaire S de symboles, appelé *signature* et partitionné en deux sous-ensembles : l'ensemble F des symboles de *fonction* et l'ensemble R des symboles de *relation*. A chaque symbole de fonction et de relation est attaché un entier n positif ou nul, son *arité*. Un symbole n -aire est un symbole d'arité n . Un symbole de fonction 0-aire est appelé *constante*. Fixons nous alors une signature $S = F \cup R$ pour tout ce chapitre 2.

Un *terme* ou *S-terme*, est un mot construit sur $L \cup S \cup V$, de l'une des deux formes suivantes

$$x, ft_1 \dots t_n,$$

avec x pris dans V , f un symbole de fonction n -aire pris dans F et les t_i des termes de tailles plus petites que celui qui est en train d'être défini.

Une *formule* ou *S-formule* est un mot construit sur $L \cup S \cup V$ de l'une des onze formes suivantes

$$\begin{aligned} s &= t, rt_1 \dots t_n, \text{vrai}, \text{faux}, \\ \neg\varphi, (\varphi \wedge \psi), (\varphi \vee \psi), (\varphi \rightarrow \psi), (\varphi \leftrightarrow \psi), \quad (1) \\ &(\forall x \varphi), (\exists x \varphi), \end{aligned}$$

avec $x \in V$, s, t et t_i des termes, r un symbole de relation n -aire pris dans R et φ et ψ des formules de tailles plus petites que celle qui est en train d'être définie. Les formules de la première ligne de (1) sont dites *atomiques*, et à *plat* si elles sont de l'une des formes suivantes

$$\text{vrai}, \text{faux}, x_0 = f x_1 \dots x_n, x_0 = x_1, r x_1 \dots x_n,$$

où les x_i sont des variables éventuellement non distinctes prises dans V , f un symbole de fonction n -aire

pris dans F et r un symbole de relation n -aire pris dans R . Une *équation* ou *S-équation* est une formule de la forme $s = t$ avec s et t des termes. Par abus de langage, une *relation* ou *S-relation* est une formule de la forme $rt_1 \dots t_n$ avec r un symbole de relation n -aire pris dans R et les t_i des termes. L'ensemble des termes et formules forme un *langage du premier ordre avec égalité*.

On rappelle qu'une occurrence d'une variable x dans une formule est *liée* si elle se produit à l'intérieur d'une sous-formule de la forme $(\forall x \varphi)$ ou $(\exists x \varphi)$. Elle est *libre* dans le cas contraire. Les *variables libres d'une formule* sont celles qui ont au moins une occurrence libre dans cette formule. Si φ est une formule, alors on note $\text{var}(\varphi)$ l'ensemble des variables libres de φ . Une *proposition* est une formule sans variables libres.

La syntaxe des expressions étant contraignante, on se permet d'ajouter des parenthèses et d'en enlever quand il n'y a pas d'ambiguïtés. Nous ne distinguons pas également deux formules qui peuvent être rendues égales moyennant les transformations suivantes de sous-formules

$$\begin{aligned} \varphi \wedge \psi &\implies \psi \wedge \varphi, & (\varphi \wedge \psi) \wedge \phi &\implies \varphi \wedge (\psi \wedge \phi), \\ \varphi \wedge \text{vrai} &\implies \varphi, & \varphi \vee \text{faux} &\implies \varphi. \end{aligned}$$

Si I est l'ensemble $\{i_1, \dots, i_n\}$, alors on appelle *conjonction* de formules et on écrit $\bigwedge_{i \in I} \varphi_i$, toute formule de la forme $\varphi_{i_1} \wedge \varphi_{i_2} \wedge \dots \wedge \varphi_{i_n} \wedge \text{vrai}$. En particulier, si $I = \emptyset$, alors la conjonction $\bigwedge_{i \in I} \varphi_i$ se réduit à la formule *vrai*. On note PL l'ensemble des conjonctions de formules à plat et AT l'ensemble des conjonctions de formules atomiques. Un ensemble Ψ de formules est *fermé pour la conjonction* si pour toutes formules φ et ϕ de Ψ , la formule $\varphi \wedge \phi$ est un élément de Ψ . Enfin, si I est un ensemble fini, alors on note $\text{Card}(I)$ sa cardinalité, c'est-à-dire, le nombre de ses éléments.

2.1 Modèle

Un *modèle* ou *S-modèle* est un triplet $M = (\mathcal{M}, \mathcal{F}, \mathcal{R})$, où \mathcal{M} est un ensemble non vide contenant les *individus* du modèle M et où \mathcal{F} et \mathcal{R} sont des ensembles de fonctions et de relations dans l'ensemble \mathcal{M} , indicés par les éléments de S .

Soit $M = (\mathcal{M}, \mathcal{F}, \mathcal{R})$ un modèle. Une *M-formule* φ est une formule construite sur la signature $S \cup \mathcal{M}$ au lieu de S , en considérant les éléments de \mathcal{M} comme des symboles de fonction d'arité 0. Si pour chaque variable libre x de φ , on remplace toute occurrence de x par un même élément de \mathcal{M} , alors on obtient une *M-formule* appelée *instanciation* de φ par des individus de M .

Soit φ est une formule, on dit que φ est *vraie dans* M et on écrit

$$M \models \varphi, \quad (2)$$

pour signifier que, pour toute instanciation φ' de φ par des individus de M , l'ensemble \mathcal{M} a la propriété exprimée par φ' , lorsqu'on interprète les symboles de fonction et de relation de φ' par les fonctions et relations correspondantes dans \mathcal{M} , et lorsqu'on attribue aux symboles logiques leur sens habituel.

Terminons par une notation commode. Soient $\bar{x} = x_1 \dots x_n$ un mot sur \mathbf{v} et $\bar{i} = i_1 \dots i_n$ un mot sur \mathcal{M} ou \mathbf{v} de la même longueur que \bar{x} . Soient $\varphi(\bar{x})$ une *M-formule*. On désigne par $\varphi(\bar{i})$, la *M-formule* obtenue en remplaçant dans $\varphi(\bar{x})$ chaque occurrence libre de x_j par i_j .

2.2 Théorie

Une *théorie* ou *S-théorie* T est un ensemble de propositions éventuellement infini appelées *axiomes*. On dit que le modèle M est un modèle de T si pour tout élément φ de T , $M \models \varphi$. Si φ est une formule, on écrit $T \models \varphi$ si pour tout modèle M de T , $M \models \varphi$. On dit que les formules φ et ψ sont *équivalentes dans* T si $T \models \varphi \leftrightarrow \psi$.

2.3 Quantificateurs vectoriels : $\exists?$, $\exists!$

Soient M un modèle et T une théorie. Soient $\bar{x} = x_1 \dots x_n$ et $\bar{y} = y_1 \dots y_n$ deux mots sur \mathbf{v} de même longueur. Soient ψ , ϕ , φ et $\varphi(\bar{x})$ des *M-formules*. On écrit

$$\begin{aligned} \exists \bar{x} \varphi &\quad \text{pour } \exists x_1 \dots \exists x_n \varphi, \\ \forall \bar{x} \varphi &\quad \text{pour } \forall x_1 \dots \forall x_n \varphi, \\ \exists? \bar{x} \varphi(\bar{x}) &\quad \text{pour } \forall \bar{x} \forall \bar{y} \varphi(\bar{x}) \wedge \varphi(\bar{y}) \rightarrow \bigwedge_{i \in \{1, \dots, n\}} x_i = y_i, \\ \exists! \bar{x} \varphi &\quad \text{pour } (\exists \bar{x} \varphi) \wedge (\exists? \bar{x} \varphi). \end{aligned}$$

Le mot \bar{x} , qui peut éventuellement être le mot vide ε , est appelé *vecteur de variables*. Les formules $\exists? \varepsilon \varphi$ et $\exists! \varepsilon \varphi$ sont respectivement équivalentes à *vrai* et à φ dans tout modèle M . Notons que ces quantificateurs ne sont que des notations commodes qui peuvent s'exprimer au premier ordre.

2.4 Quantificateur zéro-infini : $\exists_o^{\Psi(u)}$

Soient M un modèle et T une théorie. Soit $\Psi(u)$ un ensemble de formules ayant au plus une variable libre u . Soient φ , φ_j des *M-formules*.

Définition 2.4.1 On écrit

$$M \models \exists_o^{\Psi(u)} x \varphi(x), \quad (3)$$

si pour toute instanciation $\exists x \varphi'(x)$ de $\exists x \varphi(x)$ par des individus de M une des propriétés suivantes est vérifiée

- l'ensemble des individus i de M tels que $M \models \varphi'(i)$, est infini

- pour tout sous ensemble fini $\{\psi_1(u), \dots, \psi_n(u)\}$ d'éléments de $\Psi(u)$, l'ensemble des individus i de M tels que $M \models \varphi'(i) \wedge \bigwedge_{j \in \{1, \dots, n\}} \neg \psi_j(i)$ est infini.

On écrit $T \models \exists_o^\Psi(u) x \varphi(x)$, si pour tout modèle M de T on a $M \models \exists_o^\Psi(u) x \varphi(x)$.

2.5 Théories zéro-infini-décomposable [5]

Définition 2.5.1 Une théorie T ayant au moins un modèle est dite zéro-infini-décomposable, s'il existe un ensemble $\Psi(u)$ de formules, ayant au plus une variable libre u , un ensemble A de formules fermé pour la conjonction, un ensemble A' de formules de la forme $\exists \bar{x} \alpha$ avec $\alpha \in A$, et un sous-ensemble A'' de A tels que

1. Toute formule de la forme $\exists \bar{x} \alpha \wedge \psi$, avec $\alpha \in A$ et ψ une formule quelconque, est équivalente dans T à une formule de la forme

$$\exists \bar{x}' \alpha' \wedge (\exists \bar{x}'' \alpha'' \wedge (\exists \bar{x}''' \alpha''' \wedge \psi)),$$

avec $\exists \bar{x}' \alpha' \in A'$, $\alpha'' \in A''$, $\alpha''' \in A$ et $T \models \forall \bar{x}'' \alpha'' \rightarrow \exists! \bar{x}''' \alpha'''$,

2. Si $\exists \bar{x}' \alpha' \in A'$, alors $T \models \exists ? \bar{x}' \alpha'$ et pour toute variable libre y dans $\exists \bar{x}' \alpha'$, au moins une des propriétés suivantes est vérifiée
 - $T \models \exists ? y \bar{x}' \alpha'$,
 - il existe $\psi(u) \in \Psi(u)$ tel que $T \models \forall y (\exists \bar{x}' \alpha') \rightarrow \psi(y)$,
3. Si $\alpha'' \in A''$ alors
 - la formule $\neg \alpha''$ est équivalente dans T à une formule de la forme $\bigvee_{i \in I} \alpha_i$ avec $\alpha_i \in A$,
 - pour tout x'' , la formule $\exists x'' \alpha''$ est équivalente dans T à une formule qui appartient à A'' ,
 - pour tout variable x'' , $T \models \exists_o^\Psi(u) x'' \alpha''$.
4. Toute conjonction de formules à plat est équivalente dans T à une disjonction d'éléments de A .
5. Si la formule $\exists \bar{x}' \alpha' \wedge \alpha''$ avec $\exists \bar{x}' \alpha' \in A'$ et $\alpha'' \in A''$ n'a pas de variables libres, alors $\bar{x} = \varepsilon$, et α' et α'' sont des éléments de $\{\text{vrai}, \text{faux}\}$.

Théorème 2.5.2 Si T est zéro-infini-décomposable alors T est complète.

La preuve de ce théorème est donnée en détail dans [5].

3 Extension en arbres de théories du premier ordre

3.1 Théorie des arbres éventuellement infinis

Soit S une signature contenant uniquement un ensemble infini F de symboles de fonction. Michael Maher a introduit la S -théorie des arbres éventuellement

infini [7]. L'axiomatisation de cette S -théorie est l'ensemble des S -propositions de l'une des formes suivantes :

- 1 $\forall \bar{x} \forall \bar{y} f \bar{x} = f \bar{y} \rightarrow \bigwedge_i x_i = y_i$,
- 2 $\forall \bar{x} \forall \bar{y} \neg f \bar{x} = g \bar{y}$,
- 3 $\forall \bar{x} \exists ! \bar{z} \bigwedge_i z_i = t_i(\bar{z}, \bar{x})$,

avec $f, g \in F$, x, y, z des variables, \bar{x} un vecteur de variables x_i , \bar{y} un vecteur de variables y_i , \bar{z} un vecteur de variables z_i toutes distinctes et $t_i(\bar{x}, \bar{z})$ un S -terme qui commence par un élément de F suivi de variables prises dans \bar{x} ou \bar{z} .

Le premier axiome est appelé *axiome de l'explosion*, le deuxième *axiome de conflit de symboles* et le troisième *axiome de solution unique*.

3.2 Axiomatisation de T^*

Soit maintenant une signature S contenant un ensemble F de symboles de fonction et un ensemble R de symboles de relation. Soit F^* un ensemble infini de symboles de fonction contenant l'ensemble F , et soit $R^* = R \cup \{p\}$ un ensemble de symboles de relation contenant R ainsi qu'un symbole de relation p d'arité 1. Notons S^* la signature contenant F^* et R^* .

Soit T et une S -théorie, donnons maintenant l'axiomatisation du mélange de T avec la théorie des arbres éventuellement infinis. L'extension en arbres de la S -théorie T est la S^* -théorie T^* dont les axiomes sont l'ensemble des S^* -propositions de l'une des formes suivantes, avec \bar{x} un vecteur de variables x_i et \bar{y} un vecteur de variables y_i :

1. L'explosion : pour tout $f \in F^*$:

$$\forall \bar{x} \forall \bar{y} \neg p f \bar{x} \wedge \neg p f \bar{y} \wedge f \bar{x} = f \bar{y} \rightarrow \bigwedge_i x_i = y_i$$

2. Conflit de symboles : Soit f et g deux symboles de fonction distincts pris dans F^* :

$$\forall \bar{x} \forall \bar{y} f \bar{x} = g \bar{y} \rightarrow p f \bar{x} \wedge p g \bar{y}$$

3. Solution unique

$$\forall \bar{x} \forall \bar{y} \left(\bigwedge_i p x_i \wedge \bigwedge_j \neg p y_j \right) \rightarrow \exists ! \bar{z} \bigwedge_k (\neg p z_i \wedge z_k = t_k(\bar{x}, \bar{y}, \bar{z}))$$

où \bar{z} est un vecteur de variables distinctes z_i , $t_k(\bar{x}, \bar{y}, \bar{z})$ un S^* -terme qui commence par un symbole de fonction $f_k \in F^*$ suivi par des variables prises dans $\bar{x}, \bar{y}, \bar{z}$, de plus, si $f_k \in F$, alors le S^* -terme $t_k(\bar{x}, \bar{y}, \bar{z})$ contient au moins une variable de \bar{y} ou \bar{z}

4. Relations de R : pour tout $r \in R$,

$$\forall \bar{x} r \bar{x} \rightarrow \bigwedge_i p x_i$$

5. Opérations de F : pour tout $f \in F$,

$$\forall \bar{x} \, pf\bar{x} \leftrightarrow \bigwedge_i px_i$$

(Si f est d'arité 0 alors cet axiome s'écrit pf)

6. Eléments en dehors de T : pour tout $f \in F^* - F$,

$$\forall \bar{x} \, \neg pf\bar{x}$$

7. Existence d'au moins un élément satisfaisant p (uniquement si F ne contient pas de symboles de fonction d'arité 0) : $\exists x \, px$,
8. Extension en arbres des axiomes de T : Tout axiome obtenu en utilisant les transformations suivantes sur chaque axiome φ de T : Tant qu'il est possible remplacer toute sous-formule de φ de la forme $\exists \bar{x} \, \psi$, avec ψ une S^* -formule qui n'est pas de la forme $(\bigwedge px_i) \wedge \psi'$, par $\exists \bar{x} (\bigwedge px_i) \wedge \psi$ et toute sous formule de φ de la forme $\forall \bar{x} \, \psi$, avec ψ une S^* -formule qui n'est pas de la forme $(\bigwedge px_i) \rightarrow \psi'$, par $\forall \bar{x} (\bigwedge px_i) \rightarrow \psi$.

Notons le rôle important du symbole de relation unaire p qui permet de distinguer les éléments arborescents des éléments non arborescents dans tout mélange $T + \text{Arbres}$.

3.3 Exemples

3.3.1 Extension en arbres de la théorie vide

Soit $S = \emptyset$ une signature vide et T la S -théorie vide. Cette théorie vide a bien entendu comme modèle tout ensemble non vide quelconque. Soit maintenant $S^* = F^* \cup \{p\}$ une signature telle que F^* est un ensemble infini de symboles de fonction, tous d'arité non nulle, et p un symbole de relation unaire. L'extension en arbres de T est la S^* -théorie T^* dont l'ensemble des axiomes consiste en l'ensemble des propositions de l'une des formes suivantes

1. L'explosion : pour tout $f \in F^*$:

$$\forall \bar{x} \forall \bar{y} \, \neg pf\bar{x} \wedge \neg pf\bar{y} \wedge f\bar{x} = f\bar{y} \rightarrow \bigwedge_i x_i = y_i$$

2. Conflit de symboles : Soit f et g deux symboles de fonction distincts pris dans F^* :

$$\forall \bar{x} \forall \bar{y} \, f\bar{x} = g\bar{y} \rightarrow pf\bar{x} \wedge pg\bar{y}$$

3. Solution unique

$$\forall \bar{x} \forall \bar{y} \, (\bigwedge_i px_i) \wedge (\bigwedge_j \neg py_j) \rightarrow \exists! \bar{z} \bigwedge_k (\neg pz_k \wedge z_k = t_k(\bar{x}, \bar{y}, \bar{z}))$$

où \bar{z} est un vecteur de variables distinctes z_i , $t_k(\bar{x}, \bar{y}, \bar{z})$ un S^* -terme qui commence par un symbole de fonction $f_k \in F^*$ suivi par des variables prises dans $\bar{x}, \bar{y}, \bar{z}$,

4. Eléments en dehors de T : pour tout $f \in F^*$,

$$\forall \bar{x} \, \neg pf\bar{x}$$

5. Existence d'au moins un élément satisfaisant p :

$$\exists x \, px.$$

Utilisons maintenant l'axiome 4, pour simplifier ces axiomes. Nous allons également remplacer le symbole p par *feuille* afin de clarifier les axiomes de cette théorie et leur donner un sens clair. On obtient alors l'axiomatisation suivante

1. L'explosion : pour tout $f \in F^*$:

$$\forall \bar{x} \forall \bar{y} \, f\bar{x} = f\bar{y} \rightarrow \bigwedge_i x_i = y_i$$

2. Conflit de symboles : Soit f et g deux symboles de fonction distincts pris dans F^* :

$$\forall \bar{x} \forall \bar{y} \, f\bar{x} = g\bar{y}$$

3. Solution unique

$$\forall \bar{x} \exists! \bar{z} \bigwedge_k z_k = t_k(\bar{x}, \bar{z})$$

où \bar{z} est un vecteur de variables distinctes z_i , $t_k(\bar{x}, \bar{z})$ un S^* -terme qui commence par un symbole de fonction $f_k \in F^*$ suivi par des variables prises dans \bar{x} ou \bar{z} ,

4. Eléments en dehors de T : pour tout $f \in F^*$,

$$\forall \bar{x} \, \neg \text{feuille } f\bar{x}$$

5. Existence d'au moins un élément satisfaisant *feuille* :

$$\exists x \, \text{feuille } x.$$

Cette axiomatisation est l'axiomatisation de la théorie des arbres finis ou infinis de M. Maher [7], construite sur l'ensemble F^* et augmentée du symbole de relation unaire *feuille*. Cependant, cette axiomatisation force tout modèle de T^* à contenir au moins un arbre réduit à une feuille. Cette dernière restriction provient du fait que cette théorie est une extension en arbre de la théorie vide dont chaque modèle a au moins un individu i (voir la définition de modèle).

3.3.2 Extension en arbres T_{ord}^* de l'ordre dense sans extrêmes T_{ord}

Soient F un ensemble de symboles de fonction vide et R un ensemble de symboles de relation contenant uniquement le symbole de relation $<$ d'arité 2. Si t_1 et t_2 sont des termes, alors on écrit $t_1 < t_2$ pour

$< (t_1, t_2)$. Soit T_{ord} la théorie de l'ordre strict, total, dense et sans extrêmes, de signature $S = F \cup R$ et dont les axiomes sont l'ensemble des propositions suivantes

- 1 $\forall x \neg x < x,$
- 2 $\forall x \forall y \forall z (x < y \wedge y < z) \rightarrow x < z,$
- 3 $\forall x \forall y x < y \vee x = y \vee y < x,$
- 4 $\forall x \forall y x < y \rightarrow (\exists z x < z \wedge z < y),$
- 5 $\forall x \exists y x < y,$
- 6 $\forall x \exists y y < x.$

Soient maintenant F^* un ensemble infini de symboles de fonction tous d'arité non nulle et $R^* = \{<, p\}$ un ensemble de symboles de relation contenant le symbole $<$ ainsi qu'un symbole de relation unaire p . Soit alors S^* la signature $F^* \cup R^*$. En utilisant les transformations de la section 3.2, l'axiomatisation de l'extension en arbre de la théorie T_{ord} de l'ordre total, strict, dense et sans extrêmes, est la S^* -théorie T_{ord}^* dont les axiomes sont l'ensemble infini de propositions de l'une des formes suivantes

- 1 $\forall \bar{x} \forall \bar{y} \neg p f \bar{x} \wedge \neg p f \bar{y} \wedge f \bar{x} = f \bar{y} \rightarrow \bigwedge_i x_i = y_i$
- 2 $\forall \bar{x} \forall \bar{y} f \bar{x} = g \bar{y} \rightarrow p f \bar{x} \wedge p g \bar{y}$
- 3 $\forall \bar{x} \forall \bar{y} (\bigwedge_i p x_i) \wedge (\bigwedge_j \neg p y_j) \rightarrow$
 $\exists! \bar{z} \bigwedge_k (\neg p z_k \wedge z_k = t_k(\bar{x}, \bar{y}, \bar{z}))$
- 4 $\forall x \forall y x < y \rightarrow (p x \wedge p y),$
- 5 $\forall \bar{x} \neg p f \bar{x},$
- 6 $\exists x p x,$
- 7 $\forall x p x \rightarrow \neg x < x,$
- 8 $\forall x \forall y \forall z p x \wedge p y \wedge p z \rightarrow ((x < y \wedge y < z) \rightarrow x < z),$
- 9 $\forall x \forall y (p x \wedge p y) \rightarrow (x < y \vee x = y \vee y < x),$
- 10 $\forall x \forall y (p x \wedge p y) \rightarrow (x < y \rightarrow$
 $(\exists z p z \wedge x < z \wedge z < y)),$
- 11 $\forall x p x \rightarrow (\exists y p y \wedge x < y),$
- 12 $\forall x p x \rightarrow (\exists y p y \wedge y < x),$

où f et g sont des symboles de fonction distincts pris dans F^* , x, y, z des variables, \bar{x} un vecteur de variables x_i , \bar{y} un vecteur de variables y_i , \bar{z} un vecteur de variables z_i toutes distinctes et $t_i(\bar{x}, \bar{y}, \bar{z})$ un terme qui commence par un élément de F^* suivi de variables prises dans \bar{x}, \bar{y} ou \bar{z} .

Du fait de l'axiome 5, et en remplaçant le symbole de relation p par le symbole $<$, cette axiomatisation se simplifie en

- 1 $\forall \bar{x} \forall \bar{y} f \bar{x} = f \bar{y} \rightarrow \bigwedge_i x_i = y_i$
- 2 $\forall \bar{x} \forall \bar{y} \neg (f \bar{x} = g \bar{y})$
- 3 $\forall \bar{x} \exists! \bar{z} \bigwedge_k z_k = t_k(\bar{x}, \bar{z})$
- 4 $\forall x \forall y x < y \rightarrow (p x \wedge p y),$
- 5 $\forall \bar{x} \neg p f \bar{x},$
- 6 $\exists x p x,$
- 7 $\forall x p x \rightarrow \neg x < x,$
- 8 $\forall x \forall y \forall z p x \wedge p y \wedge p z \rightarrow ((x < y \wedge y < z) \rightarrow x < z),$
- 9 $\forall x \forall y (p x \wedge p y) \rightarrow (x < y \vee x = y \vee y < x),$
- 10 $\forall x \forall y (p x \wedge p y) \rightarrow (x < y \rightarrow (\exists z p z \wedge x < z \wedge z < y)),$
- 11 $\forall x p x \rightarrow (\exists y p y \wedge x < y),$
- 12 $\forall x p x \rightarrow (\exists y p y \wedge y < x),$

où f et g sont des symboles de fonction distincts pris dans F^* , x, y, z des variables, \bar{x} un vecteur de variables x_i , \bar{y} un vecteur de variables y_i , \bar{z} un vecteur de variables z_i toutes distinctes et $t_i(\bar{z}, \bar{x})$ un terme qui commence par un élément de F^* suivi de variables prises dans \bar{x} ou \bar{z} .

Cette axiomatisation a pour modèle tout construction d'arbres sur un ensemble quelconque muni d'une

relation d'ordre strict, total, dense et sans extrêmes. Bien entendu, la méthode donnée dans ce chapitre pour générer les axiomes d'une extension T^* de T est généraliste, il n'est pas écarté que l'on ait à simplifier d'avantage l'axiomatisation obtenue dans des cas particuliers, comme il a été fait pour l'extension de la théorie vide et l'extension de T_{ord} .

3.4 Complétude de T^*

Fixons-nous maintenant pour toute cette section une signature S contenant un ensemble F de symboles de fonction et un ensemble R de symboles de relation, ainsi qu'une signature S^* contenant

- un ensemble infini $F^* = F \cup F_A$ où F_A est un ensemble infini de symboles de fonction, tous d'arité non nulle, et disjoint de F ,
- un ensemble $R^* = R \cup \{p\}$ de symboles de relation, contenant R ainsi qu'un symbole de relation p d'arité 1.

Fixons-nous également une S -théorie T ainsi que son extension en arbres T^* .

Supposons que les variables de V soient ordonnées par un ordre total, strict, dense et sans extrêmes, noté \succ .

Définition 3.4.1 On appelle représentant d'une S -équation α la plus grande variable x dans α , suivant l'ordre \succ , telle que $T \models \exists! x \alpha$.

Exemple 3.4.2 Soit x, y et z des variables telles que $x \succ y \succ z$. Reprenons l'exemple de la théorie Ra des rationnels additifs définie dans le chapitre 2. La variable x est représentant de l'équation $2.x + 3.y = 1 + 2.z$ car x est la plus grande variable et $Ra \models \exists! x 2.x + 3.y = 1 + 2.z$.

Définition 3.4.3 Une conjonction de S -formules atomiques α est dite formatée dans T si

- α ne contient pas de sous-formules de la forme $f_1 = f_2$ ou $r f_1 \dots f_n$ ou $y = x$, avec f_i des constantes de F , $r \in R$ et $x \succ y$,
- chaque S -équation de α a un représentant distinct qui n'a pas d'occurrence dans d'autres S -équations ou S -relations de α ,
- si α' est la conjonction des S -équations de α , alors pour tout $x \in \text{var}(\alpha')$ on a $T \models \exists? x \alpha'$.

Nous allons maintenant introduire une famille de théories T_i dont les propriétés nous seront très utiles pour montrer la zéro-infini-décomposabilité de leur extension en arbres, c'est-à-dire, T_i^* .

Définition 3.4.4 La théorie T est dite flexible si pour toute conjonction α de S -équations et pour toute conjonction β de S -relations

1. $\alpha \wedge \beta$ est équivalente dans T à une conjonction formatée de formules atomiques,
2. la S -formule $\neg\beta$ est équivalente dans T à une disjonction de S -équations et de S -relations,
3. pour tout $x \in V$
 - la S -formule $\exists x \beta$ est équivalente dans T à faux, ou à une conjonction de S -relations,
 - pour tout $x \in V$, on a $T \models \exists_o^{\{f_{aux}\}} x \beta$.

Annonçons maintenant notre résultat principal

Théorème 3.4.5 *Si T est flexible alors T^* est une théorie zéro-infini-décomposable est donc complète.*

Ce résultat est assez important du moment qu'il nous évite d'étudier la théorie T^* au profil de la théorie T . En effet, nous disposons maintenant d'une condition suffisante de complétude de toute extension en arbres T^* d'une théorie T du premier ordre et qui donne des conditions sur T et uniquement sur T pour que le mélange $T + \text{Arbre}$ soit complet. Qu'en ait-il maintenant de l'algorithme de décision dans T^* ?

4 Décision de propositions dans T^*

Fixons nous une signature S^* pour toute cette section et travaillons uniquement sur des théories de signature S^* . On peut se permettre alors d'omettre le préfixe S^* au début de mots : formules, équations, théories et modèles. Soit alors T^* une théorie zéro-infini-décomposable. Les ensembles $\Psi(u)$, A , A' et A'' sont donc connus et fixés tout au long de cette section.

4.1 Formule normalisée

Définition 4.1.1 *Une formule normalisée φ de profondeur $d \geq 1$ est une formule de la forme*

$$\neg(\exists \bar{x} \alpha \wedge \bigwedge_{i \in I} \varphi_i),$$

avec I un ensemble fini (éventuellement vide), $\alpha \in PL$ et tous les φ_i sont des formules normalisées de profondeur d_i avec $d = 1 + \max\{0, d_1, \dots, d_n\}$.

Propriété 4.1.2 *Toute formule φ est équivalente dans la théorie vide à une formule normalisée de profondeur $d \geq 1$.*

Il est facile de transformer toute formule en une formule normalisée; il suffit par exemple de suivre les étapes suivantes

1. Introduire un supplément d'équations et de variables quantifiées pour transformer les conjonctions de formules atomiques en conjonctions de formules à plat.

2. Exprimer les quantificateurs, constantes et symboles logiques avec \neg , \wedge et \exists , en utilisant les transformations de sous-formules suivantes

$$\begin{aligned} (\varphi \vee \phi) &\implies \neg(\neg\varphi \wedge \neg\phi), \\ (\varphi \rightarrow \phi) &\implies \neg(\varphi \wedge \neg\phi), \\ (\varphi \leftrightarrow \phi) &\implies (\neg(\varphi \wedge \neg\phi) \wedge \neg(\phi \wedge \neg\varphi)), \\ (\forall x \varphi) &\implies \neg(\exists x \neg\varphi). \end{aligned}$$

3. Si la formule φ obtenue ne commence pas par le symbole \neg , alors la remplacer par $\neg(\exists \varepsilon \text{ vrai} \wedge \neg\varphi)$.
4. Nommer les variables quantifiées par des noms distincts et différents des noms des variables libres.
5. Remonter les quantifications au dessus des conjonctions, c'est-à-dire $\varphi \wedge (\exists \bar{x} \psi)$ ou $(\exists \bar{x} \psi) \wedge \varphi$, devient $\exists \bar{x} \varphi \wedge \psi$ car les variables libres de φ sont différentes de celles de \bar{x} .
6. Grouper les variables quantifiées dans un vecteur de variables, c'est-à-dire $\exists \bar{x}(\exists \bar{y} \varphi)$ ou $\exists \bar{x} \exists \bar{y} \varphi$ devient $\exists \bar{x} \bar{y} \varphi$.
7. Insérer des vecteurs vides et des formules de la forme *vrai* pour obtenir la formule normalisée finale en utilisant les transformations suivantes de sous-formules

$$\neg(\bigwedge_{i \in I} \neg\varphi_i) \implies \neg(\exists \varepsilon \text{ vrai} \wedge \bigwedge_{i \in I} \neg\varphi_i), \quad (4)$$

$$\neg(\alpha \wedge \bigwedge_{i \in I} \neg\varphi_i) \implies \neg(\exists \varepsilon \alpha \wedge \bigwedge_{i \in I} \neg\varphi_i), \quad (5)$$

$$\neg(\exists \bar{x} \bigwedge_{j \in J} \neg\varphi_j) \implies \neg(\exists \bar{x} \text{ vrai} \wedge \bigwedge_{j \in J} \neg\varphi_j), \quad (6)$$

avec $\alpha \in PL$, I un ensemble fini (éventuellement vide) et J un ensemble fini non vide.

Si la formule de départ ne contient pas le symbole logique \leftrightarrow , alors cette transformation est linéaire, c'est-à-dire qu'il existe une constante k telle que $n_2 \leq kn_1$, avec n_1 la taille de la formule de départ et n_2 la taille de la formule normalisée finale. Ces transformations ne dépendent pas du fait que la théorie soit décomposable ou pas, ils sont valides dans la théorie vide et donc valides dans toute théorie du premier ordre.

Exemple 4.1.3 *Soit $f \in F$ un symbole de fonction d'arité 2. Appliquons les étapes précédentes pour transformer la formule suivante en une formule normalisée équivalente dans T*

$$(fuv = fwu \wedge (\exists x u = x)) \vee (\exists u \forall w u = fvw).$$

Notons que la formule ne commence pas par le symbole \neg et que les variables u et w sont libres dans $fuv = fwu \wedge (\exists x u = x)$ et liées dans $\exists u \forall w u = fvw$.

Etape 1 : Transformons les équations en équations à plat. La formule précédente est équivalente dans T à

$$(\exists u_1 u_1 = fuv \wedge u_1 = fwu \wedge (\exists x u = x)) \vee (\exists u \forall w u = fvw).$$

Etape 2 : Exprimons maintenant le quantificateur \forall en utilisant \neg , \wedge et \exists . La formule précédente est équivalente dans T à

$$(\exists u_1 u_1 = fuv \wedge u_1 = fwu \wedge (\exists x u = x)) \vee (\exists u \neg(\exists w \neg(u = fvw))).$$

Transformons également les formules contenant le symbole \vee en utilisant \neg , \wedge et \exists . La formule précédente est équivalente dans T à

$$\neg(\neg(\exists u_1 u_1 = fuv \wedge u_1 = fwu \wedge (\exists x u = x)) \wedge \neg(\exists u \neg(\exists w \neg(u = fvw)))).$$

Etape 3 : La formule commence par le symbole \neg , alors on passe à l'étape 4.

Etape 4 : Les variables quantifiées u et w dans $(\exists u \neg(\exists w \neg(u = fvw)))$ doivent être renommées. La formule précédente est équivalente dans T à

$$\neg(\neg(\exists u_1 u_1 = fuv \wedge u_1 = fwu \wedge (\exists x u = x)) \wedge \neg(\exists u_2 \neg(\exists w_1 \neg(u_2 = fvw_1)))).$$

Etape 5 : En remontant la quantification $\exists x$, la formule précédente est équivalente dans T à

$$\neg(\neg(\exists u_1 \exists x u_1 = fuv \wedge u_1 = fwu \wedge u = x) \wedge \neg(\exists u_2 \neg(\exists w_1 \neg(u_2 = fvw_1)))).$$

Etape 6 : Groupons maintenant les variables x et u_1 dans un quantificateur vectoriel. La formule précédente est équivalente dans T à

$$\neg(\neg(\exists u_1 x u_1 = fuv \wedge u_1 = fwu \wedge u = x) \wedge \neg(\exists u_2 \neg(\exists w_1 \neg(u_2 = fvw_1)))).$$

Etape 7 : d'après la règle (4), la formule précédente est équivalente dans T à

$$\neg \left[\exists \varepsilon \text{ vrai} \wedge \left[\begin{array}{l} \neg(\exists u_1 x u_1 = fuv \wedge u_1 = fwu \wedge u = x) \wedge \\ \neg(\exists u_2 \neg(\exists w_1 \neg(u_2 = fvw_1))) \end{array} \right] \right].$$

En utilisant la règle (5), la formule précédente est équivalente dans T à

$$\neg \left[\exists \varepsilon \text{ vrai} \wedge \left[\begin{array}{l} \neg(\exists u_1 x u_1 = fuv \wedge u_1 = fwu \wedge u = x) \wedge \\ \neg(\exists u_2 \neg(\exists w_1 \neg(\exists \varepsilon u_2 = fvw_1))) \end{array} \right] \right].$$

Finalement, en utilisant la règle (6) la formule précédente est équivalente dans T à

$$\neg \left[\exists \varepsilon \text{ vrai} \wedge \left[\begin{array}{l} \neg(\exists u_1 x u_1 = fuv \wedge u_1 = fwu \wedge u = x) \wedge \\ \neg(\exists u_2 \text{ vrai} \wedge \neg(\exists w_1 \text{ vrai} \wedge \neg(\exists \varepsilon u_2 = fvw_1))) \end{array} \right] \right],$$

Cette formule est une formule normalisée de profondeur égale à quatre.

4.2 Formule de travail

Définition 4.2.1 Une formule de travail φ de profondeur $d \geq 1$ est une formule de la forme

$$\neg(\exists \bar{x} \alpha \wedge \bigwedge_{i \in I} \varphi_i),$$

avec I un ensemble fini éventuellement vide, $\alpha \in A$ et tous les φ_i sont des formules de travail de profondeur d_i avec $d = 1 + \max\{0, d_1, \dots, d_n\}$.

Propriété 4.2.2 Toute formule est équivalente dans T à une conjonction de formules de travail.

Définition 4.2.3 Une formule résolue est une formule de la forme $\neg(\exists \bar{x}' \alpha' \wedge \alpha'' \wedge \bigwedge_{i \in I} \neg(\exists \bar{y}'_i \beta'_i))$, avec I un ensemble fini éventuellement vide, $\exists \bar{x}' \alpha' \in A'$, $\alpha'' \in A''$, $\exists \bar{y}'_i \beta'_i \in A'$ pour tout $i \in I$, α' et α'' sont différents de la formule faux et tous les β'_i sont différents des formules vrai et faux.

Propriété 4.2.4 Soit φ une conjonction de formules résolues sans variables libres. La conjonction φ est soit la formule vrai soit la formule $\bigwedge \neg \text{vrai}$.

4.3 Règles de réécriture

Nous présentons maintenant les règles de réécriture qui transforment une conjonction de formules de travail quelconques en une conjonction de formules résolues, équivalente dans T . Appliquer la règle $p_1 \Rightarrow p_2$ à la formule de travail p , signifie remplacer dans p une sous-formule p_1 par la formule p_2 , en considérant le connecteur \wedge associatif et commutatif.

$$(1) \neg \left[\begin{array}{l} \exists \bar{x} \alpha \wedge \varphi \wedge \\ \neg(\exists \bar{y} \text{ vrai}) \end{array} \right] \Rightarrow \text{vrai}$$

$$(2) \neg \left[\begin{array}{l} \exists \bar{x} \alpha \wedge \text{faux} \wedge \varphi \end{array} \right] \Rightarrow \text{vrai}$$

$$(3) \neg \left[\begin{array}{l} \exists \bar{x} \alpha \wedge \\ \bigwedge_{i \in I} \neg(\exists \bar{y}_i \beta_i) \end{array} \right] \Rightarrow \neg \left[\begin{array}{l} \exists \bar{x}' \alpha' \wedge \alpha'' \wedge \\ \bigwedge_{i \in I} \neg(\exists \bar{x}'' \bar{y}_i \alpha''' \wedge \beta_i) \end{array} \right]$$

$$(4) \neg \left[\begin{array}{l} \exists \bar{x} \alpha \wedge \varphi \wedge \\ \neg(\exists \bar{y}' \beta' \wedge \beta'') \end{array} \right] \Rightarrow \left[\begin{array}{l} \neg(\exists \bar{x} \alpha \wedge \varphi \wedge \neg(\exists \bar{y}' \beta')) \wedge \\ \bigwedge_{i \in I} \neg(\exists \bar{x} \bar{y}' \alpha \wedge \beta' \wedge \beta'_i \wedge \varphi) \end{array} \right]$$

$$(5) \neg \left[\begin{array}{l} \exists \bar{x} \alpha \wedge \\ \bigwedge_{i \in I} \neg(\exists \bar{y}'_i \beta'_i) \end{array} \right] \Rightarrow \neg \left[\begin{array}{l} \exists \bar{x}' \alpha' \wedge \alpha''_* \\ \bigwedge_{i \in I'} \neg(\exists \bar{y}'_i \beta'_i) \end{array} \right]$$

$$(6) \neg \left[\begin{array}{l} \exists \bar{x} \alpha \wedge \varphi \wedge \\ \neg \left[\begin{array}{l} \exists \bar{y}' \beta' \wedge \beta'' \wedge \\ \bigwedge_{i \in I} \neg(\exists \bar{z}'_i \delta'_i) \end{array} \right] \end{array} \right] \Rightarrow \left[\begin{array}{l} \neg(\exists \bar{x} \alpha \wedge \varphi \wedge \neg(\exists \bar{y}' \beta' \wedge \beta'')) \wedge \\ \bigwedge_{i \in I} \neg(\exists \bar{x} \bar{y}' \bar{z}_i \alpha \wedge \beta' \wedge \beta'' \wedge \delta'_i \wedge \varphi) \end{array} \right]$$

avec α un élément de A , φ une conjonction de formules de travail et I un ensemble fini éventuellement vide. Dans la règle (3), la formule $\exists \bar{x} \alpha$ est équivalente dans T à une formule décomposée de la forme $\exists \bar{x}' \alpha' \wedge (\exists \bar{x}'' \alpha'' \wedge (\exists \bar{x}''' \alpha'''))$ avec $\exists \bar{x}' \alpha' \in A'$, $\alpha'' \in A''$, $\alpha''' \in A$, $T \models \forall \bar{x}'' \alpha'' \rightarrow \exists! \bar{x}''' \alpha'''$ et $\exists \bar{x}''' \alpha'''$ différent de $\exists \varepsilon \text{ vrai}$. Tous les β_i sont des éléments de A . Dans la règle (4), la formule $\exists \bar{x} \alpha$ est équivalente dans T à une formule décomposée de la forme $\exists \bar{x}' \alpha' \wedge (\exists \bar{x}'' \alpha'' \wedge (\exists \varepsilon \text{ vrai}))$ avec $\exists \bar{x}' \alpha' \in A'$ et $\alpha'' \in A''$. La formule $\exists \bar{y}' \beta'$ est un élément de A' . La formule β'' est élément de A'' et est différente de la formule vrai. De plus, $T \models (\neg \beta'') \leftrightarrow \bigvee_{i \in I} \beta''_i$ avec $\beta''_i \in A$. Dans la règle (5), la formule $\exists \bar{x} \alpha$ n'est pas de la forme $\exists \bar{x} \alpha_1 \wedge \alpha_2$ avec $\exists \bar{x} \alpha_1 \in A'$ et $\alpha_2 \in A''$, et

est équivalente dans T à une formule décomposée de la forme $\exists \bar{x}' \alpha' \wedge (\exists \bar{x}'' \alpha'' \wedge (\exists \varepsilon \text{ vrai}))$ avec $\exists \bar{x}' \alpha' \in A'$ et $\alpha'' \in A''$. Chaque formule $\exists \bar{y}'_i \beta'_i$ est un élément de A' . L'ensemble I' est l'ensemble des $i \in I$ tels que $\exists \bar{y}'_i \beta'_i$ ne contienne pas d'occurrences libres d'aucune variable de \bar{x}'' . De plus, $T \models (\exists \bar{x}'' \alpha'') \leftrightarrow \alpha''_*$ avec $\alpha''_* \in A''$. Dans la règle (6), $I \neq \emptyset$, $\exists \bar{y}' \beta' \in A'$, $\exists \bar{z}'_i \delta'_i \in A'$ et $\beta'' \in A''$.

Propriété 4.3.1 *Toute application répétée des règles de réécriture précédentes sur une conjonction φ de formules de travail, se termine et produit une conjonction ϕ de formules résolues équivalente à φ dans T .*

4.4 Algorithme de résolution de propositions

Ayant une proposition ψ , la résolution ou décision de ψ dans T se fait de la manière suivante

1. Transformer la formule ψ en une formule normalisée, puis en une conjonction de formules de travail φ et équivalente à ψ dans T .
2. Appliquer les règles de réécriture précédentes sur φ autant de fois que possible. A la fin, on obtient une conjonction ϕ de formules résolues.

Du fait que la transformation de la proposition ψ en conjonction de formules de travail φ soit, alors φ est une conjonction de formules de travail sans variables libres. D'après la propriété 4.3.1, l'application de nos règles sur φ produit une conjonction ϕ de formules résolues et donc une conjonction ϕ de formules résolues sans variables libres. D'après la propriété 4.2.4, ϕ est soit la formule vrai, soit la formule $\bigwedge_{i \in I} \neg \text{vrai}$. Du fait que T soit décomposable, elle a au moins un modèle et donc soit $T \models \phi$, soit $T \models \neg \phi$ et par conséquent soit $T \models \psi$, soit $T \models \neg \psi$.

5 Extension en arbres T_{ad}^* des rationnels additifs ordonnés

5.1 Axiomatisation de T_{ad}^*

Soit $F = \{+, -, 0, 1\}$ un ensemble de symboles de fonction d'arité respectives 2, 1, 0, 0. Soit $R = \{<\}$ un ensemble de symboles de relation contenant uniquement le symbole de relation binaire $<$, et soit alors S la signature $F \cup R$.

Soit a un entier positif et t_1, \dots, t_n des S -termes. Notons

- $t_1 < t_2$, le S -terme $< t_1 t_2$,
- $t_1 + t_2$, le S -terme $+ t_1 t_2$,
- $0.t_1$, le S -terme 0 ,
- $-a.t_1$, le S -terme $\underbrace{(-t_1) + \dots + (-t_1)}_a$,
- $a.t_1$, le S -terme $\underbrace{t_1 + \dots + t_1}_a$.

Soit T_{ad} la S -théorie des rationnels additifs ordonnés. L'axiomatisation de T_{ad} est l'ensemble infini de S -propositions de l'une des 14 formes suivantes

- 1 $\forall x \forall y x + y = y + x$,
- 2 $\forall x \forall y \forall z x + (y + z) = (x + y) + z$,
- 3 $\forall x x + 0 = x$,
- 4 $\forall x x + (-x) = 0$,
- 5_n $\forall x n.x = 0 \rightarrow x = 0$, $(n \neq 0)$
- 6_n $\forall x \exists ! y n.y = x$, $(n \neq 0)$
- 7 $\forall x \neg x < x$,
- 8 $\forall x \forall y \forall z (x < y \wedge y < z) \rightarrow x < z$,
- 9 $\forall x \forall y (x < y \vee x = y \vee y < x)$,
- 10 $\forall x \forall y x < y \rightarrow (\exists z x < z \wedge z < y)$,
- 11 $\forall x \exists y x < y$,
- 12 $\forall x \exists y y < x$,
- 13 $\forall x \forall y \forall z x < y \rightarrow (x + z < y + z)$,
- 14 $0 < 1$.

avec n un entier non nul. Soit maintenant F^* un ensemble infini de symboles de fonction contenant l'ensemble $\{+, -, 0, 1\}$. Soit $R^* = \{<, p\}$ un ensemble de symboles de relation contenant le symbole $<$ ainsi qu'un symbole de relation unaire p . Soit alors S^* la signature $F^* \cup R^*$. En utilisant les transformations de la section 3.2, l'axiomatisation de T_{ad}^* est l'ensemble infini des S^* -propositions de l'une des 21 formes suivantes

- 1 $\forall \bar{x} \forall \bar{y} ((\neg p f \bar{x}) \wedge (\neg p f \bar{y}) \wedge f \bar{x} = f \bar{y}) \rightarrow \bigwedge_i x_i = y_i$,
- 2 $\forall \bar{x} \forall \bar{y} f \bar{x} = g \bar{y} \rightarrow p f \bar{x} \wedge p g \bar{y}$,
- 3 $\forall \bar{x} \forall \bar{y} (\bigwedge_{i \in I} p x_i) \wedge (\bigwedge_{j \in J} \neg p y_j) \rightarrow (\exists ! \bar{z} \bigwedge_{k \in K} (\neg p z_k \wedge z_k = t_k(\bar{x}, \bar{y}, \bar{z})))$,
- 4 $\forall x \forall y x < y \rightarrow (p x \wedge p y)$,
- 5 $\forall x \forall y p x + y \leftrightarrow p x \wedge p y$,
- 6 $\forall x p - x \leftrightarrow p x$,
- 7 $\forall x \neg p h \bar{x}$,
- 8 $\forall x \forall y (p x \wedge p y) \rightarrow x + y = y + x$,
- 9 $\forall x \forall y \forall z (p x \wedge p y \wedge p z) \rightarrow x + (y + z) = (x + y) + z$,
- 10 $\forall x p x \rightarrow x + 0 = x$,
- 11 $\forall x p x \rightarrow x + (-x) = 0$,
- 12_n $\forall x p x \rightarrow (n x = 0 \rightarrow x = 0)$, $(n \neq 0)$
- 13_n $\forall x p x \rightarrow \exists ! y p y \wedge n y = x$, $(n \neq 0)$
- 14 $\forall x p x \rightarrow \neg x < x$,
- 15 $\forall x \forall y \forall z p x \wedge p y \wedge p z \rightarrow ((x < y \wedge y < z) \rightarrow x < z)$,
- 16 $\forall x \forall y (p x \wedge p y) \rightarrow (x < y \vee x = y \vee y < x)$,
- 17 $\forall x \forall y (p x \wedge p y) \rightarrow (x < y \rightarrow (\exists z p z \wedge x < z \wedge z < y))$,
- 18 $\forall x p x \rightarrow (\exists y p y \wedge x < y)$,
- 19 $\forall x p x \rightarrow (\exists y p y \wedge y < x)$,
- 20 $\forall x \forall y \forall z (p x \wedge p y \wedge p z) \rightarrow (x < y \rightarrow (x + z < y + z))$,
- 21 $0 < 1$,

avec f et g deux symboles de fonction distincts pris dans F^* , $h \in F^* - F$, x, y, z des variables, \bar{x} un vecteur de variables x_i , \bar{y} un vecteur de variables y_i , \bar{z} un vecteur de variables z_i toutes distinctes et $t_k(\bar{x}, \bar{y}, \bar{z})$ un S^* -terme qui commence par un symbole de fonction f_k élément de F^* suivi par des variables prises dans \bar{x} ou \bar{y} ou \bar{z} , de plus, si $f_k \in F$ alors $t_k(\bar{x}, \bar{y}, \bar{z})$ contient au moins une variable de \bar{y} ou \bar{z} . Une théorie similaire à celle-ci a été introduite par A. Colmerauer pour la modélisation du modèle de Prolog III [2].

Notons que la théorie des arbres et la théorie des rationnels ont des signatures non-disjointes ; en effet, les symboles $+$ et $-$ sont des constructeurs d'arbres dans la théorie des arbres et des opérations d'addition et de soustraction dans la théorie des rationnels. Notons également que T_{ad}^* n'admet pas d'élimination complète

de quantificateurs. En effet, la S^* -formule $\exists x y = fx$ avec $f \in F - \{+, -, 0, 1\}$ ne peut pas être simplifiée d'avantage ni dans T_{ad}^* ni dans la théorie des arbres.

5.2 Complétude de T_{ad}^*

Propriété 5.2.1 *La théorie T_{ad} des rationnels additifs ordonnés est une théorie flexible.*

Corollaire 5.2.2 *L'extension en arbres T_{ad}^* des rationnels additifs ordonnés T_{ad} est une théorie zéro-infini-décomposable est donc complète.*

6 Conclusion

Nous avons défini dans ce papier une idée d'extension du modèle de Prolog en donnant notamment une manière automatique pour combiner une théorie T quelconque à la théorie des arbres éventuellement infinis. Sémantiquement cette combinaison est une extension en arbres des éléments des modèles de la théorie T . Une des difficultés majeures de cette combinaison réside dans le fait que les deux théories peuvent avoir des signatures non disjointes et que la théorie des arbres n'admette pas d'élimination de quantificateurs. Après avoir défini l'axiomatisation T^* de l'extension en arbres de T nous avons introduit les théories flexibles et montré que si T est flexible alors T^* est complète et admet un algorithme de décision de propositions qui pour toute proposition donne soit *vrai* soit *faux*.

Michael Maher a montré dans [7] que la théorie des arbres éventuellement infinis n'admet pas d'élimination de quantificateurs. Ainsi l'extension en arbres T^* d'une théorie T flexible n'admet pas d'élimination de quantificateurs, ce qui ne l'empêche pas d'avoir un algorithme de décision de propositions qui utilise une décomposition locale pour une élimination partielle de quantificateurs suivie d'une distribution pour réduire la profondeur de la formule.

Cet algorithme de décision qui est idéal pour décider de la valeur de vérité de très grandes propositions peut également s'appliquer sur des formules ayant des variables libres et produit alors une conjonction de formules finales ayant des variables libres, mais en aucun il est capable de décider si cette conjonction de formules finales est toujours vraie ou toujours fausse ou ni vraie ni fausse dans T^* . Il ne donne pas de solutions pour les contraintes qui ont au moins une variable libre et n'est pas capable de détecter si une formule ayant au moins une variable libre est toujours fausse ou toujours vraie dans T^* . C'est pour toutes ces raisons que notre algorithme est appelé *algorithme de décision* et non algorithme de résolution de contraintes générales. Il serait alors intéressant de donner un algorithme de résolution de contraintes générales dans T^* qui pour toute formule φ , qui peut éventuellement contenir des

variables libres, donne soit *vrai*, soit *faux*, soit une formule ayant au moins une variable libre, n'étant équivalente ni à *vrai* ni à *faux* dans T^* et dont les solutions de ses variables libres sont exprimées d'une manière claire et explicite. Un tel algorithme demande un travail complètement différent de celui-ci avec des conditions sémantiques et syntaxiques beaucoup plus complexes que la définition de théorie flexible définie dans ce papier.

Actuellement, nous avons développé un algorithme de résolution de contraintes générales dans une combinaison de la théorie des arbres et des rationnels munis de l'addition, de la soustraction et d'une relation d'ordre dense sans extrêmes, et nous essayons de trouver un algorithme similaire dans le cas des extensions en arbres de théories. Des définitions plus intuitives et moins complexes des ensembles A' , A'' et $\Psi(u)$ sont également à prévoir.

Enfin, afin d'enrichir ce travail théorique, nous planifions avec Thom Fruehwirth [6] d'étendre le langage CHR afin qu'il puisse traiter les formules quantifiées. Pour cela, nous allons doter CHR d'un mécanisme pour manipuler nos formules normalisées, ce qui nous permettra d'effectuer des implantations rapides de nos algorithmes, de les améliorer et de se faire une idée sur l'expressivité des contraintes hybrides dans des extensions en arbres de théories.

Références

- [1] Colmerauer A. Equations and inequations on finite and infinite trees. Proc. of the Int. Conf. on the Fifth Generation of Computer Systems, pages 85-99. Tokyo, 1984.
- [2] Colmerauer A. An introduction to Prolog III. *Communication of the ACM*, 33(7) :68-90, 1990.
- [3] Colmerauer A., Dao. TBH. Expressiveness of full first-order constraints in the algebra of finite or infinite trees, *Constraints*, 8(3) :283-302. 2003.
- [4] Dao TBH. Résolution de contraintes du premier ordre dans la théorie des arbres finis ou infinis. Thèse d'informatique, Université de la Méditerranée, décembre 2000.
- [5] Djelloul K. About the combination of trees and rational numbers in a complete first-order theory. Proceeding of the 5th Int. Conf. on Frontiers of Combining Systems FroCoS 2005. LNAI, vol 3717, pp : 106-122.
- [6] Fruehwirth T., Abdelnadh S. Essentials of constraints programming. Springer Cognitive technologies.
- [7] Maher M. Complete axiomatization of the algebra of finite, rational and infinite trees. *Technical report*, IBM, 1988.